PATENT APPLICATION

Bus Architecture for System on a Chip

Inventors:

Roger May, a citizen of the United Kingdom, residing at,
30 Partridge Chase, Bicester, Oxon, United Kingdom

James Tyson, a citizen of the United Kingdom, residing at,
54 Dropmore Road, Burnham, Slough, United Kingdom

Edward Flaherty, a citizen of the United States of America, residing at,
Rimes Cottage, Kingston Bagpuize, OX13 5AY, United Kingdom

Mark Dickinson, a citizen of the United Kingdom, residing at,
Brill, Bucks, United Kingdom

Assignee:

Altera Corporation
101 Innovation Drive
San Jose, CA 95134

Entity: Large

Bus Architecture for System on a Chip

CROSS-REFERENCES TO RELATED APPLICATIONS

This application claims priority from provisional U.S. Patent Application
No. 60/211094, filed June 12, 2000 and which is incorporated by reference into this
application for all purposes.

A related application is **Attorney Docket No. 015114-053220** filed
concurrently with the present application as U.S. Patent Application No. _____ in the
names of May et al., and entitled "Setting Up Memory and Registers from a Serial
Device" and assigned to the present assignee. Another related application is **Attorney
Docket No. 015114-053230** filed concurrently with the present application as U.S. Patent
Application No. _____ in the names of May et al., and entitled "Re-configurable
Memory Map for a System on a Chip" and assigned to the present assignee.

BACKGROUND OF THE INVENTION

The present invention relates to digital systems. More specifically, the
present invention relates to a bus architecture for an integrated digital system.

Since their inception, digital systems have progressed towards higher
levels of integration. Higher integration offers several benefits to the system designer,
including lower development costs, shorter design cycles, increased performance and
generally lower power consumption. At the device level, this integration has been
achieved by the accumulation of functions once performed by multiple, individual
devices into more capable, higher density devices. Additionally, the need for design
flexibility has increased due to more challenging time-to-market pressures and changes in
system specifications.

Often at the heart of a digital system is the microprocessor, also known as
a CPU. A microprocessor is an integrated circuit implemented on a semiconductor chip,
which typically includes, among other things, an instruction execution unit, register file,
arithmetic logic unit (ALU), multiplier, etc. Microprocessors are found in digital

systems, such as personal computers for executing instructions, and can also be employed to control the operation of most digital devices.

Microprocessors have evolved, most notably, in two directions. The first is towards higher performance and the second is towards greater ease of use. The path to higher performance has produced microprocessors with wider data paths and longer instructions. Greater integration has also improved speed, as many microprocessors now incorporate on-board structures such as memory for caching. Finally, like all semiconductors, microprocessors have benefited from architectural and process enhancements, allowing higher speed through better clock rates and more efficient logic operations.

Another digital device, which has evolved over its lifetime to meet the needs of system designer is the programmable logic device (PLD). A programmable logic device is a logic element having a logic function, which is not restricted to a specific function. Rather, the logic function of a PLD is programmed by a user. PLDs provide the advantages of fixed integrated circuits with the flexibility of custom integrated circuits. Demands for greater capacity and performance have been met with larger PLD devices, architecture changes, and process improvements. Similar to microprocessors, the road to greater integration has also led to memory structures being incorporated into PLD architectures.

The traditional approach to system design involves combining a microprocessor and other off-the-shelf devices on a board, while partitioning the board's functions into the components that are best suited to perform them. While this method seems to be straightforward, it ignores the advantages to be gained by higher device-level integration. With higher device-level integration, the elimination of on-chip/off-chip delays enhances performance. Power consumption and overall manufacturing and design costs are often improved as well. Yet, integration presents problems of its own. For example, since a microprocessor will normally be clocked at a faster rate than other elements, a method and apparatus are needed to address this difference in clock speeds.

SUMMARY OF THE INVENTION

According to an embodiment of the present invention a system, which is integrated on a single chip, is disclosed. The system includes a combination of an embedded processor, reprogrammable memory, a programmable logic device (e.g. a

5   PLD) and a multiple bus architecture including bus bridges that allow communication between adjacent clock domains, yet which allow communication among the PLD, reprogrammable memory, processor, etc.

The bus architecture of the present invention, in particular, is embodied as a multiple bus master system, which allows communication among all peripherals in the

10   system, via bridges that de-couple the clock frequencies of the individual bus masters from the peripheral they are accessing. The bus architecture of the present invention, therefore, allows the system components, for example the processor peripherals, and PLD to run at their optimal speeds.

In a first aspect of the invention a digital system integrated on a

15   semiconductor chip is disclosed. The system includes one or more first bus masters coupled to a first bus in a first clock domain, a PLD coupled to a second bus in a second clock domain. A first bridge is coupled between the first and second buses and is operable to de-couple the first clock domain from the second clock domain. Additionally, one or more masters on the first bus are configured to communicate with one or more

20   slaves on the second bus. The second bus may also contain a number of masters, including the PLD.

In a second aspect of the invention, a digital system on a semiconductor chip includes a central processing unit coupled to a first bus, a programmable logic device coupled to a second bus and a bus bridge coupled between the first and second buses. In

25   this aspect of the invention, the first bus operates within a first clock domain and the second bus operates within a second clock domain.

In a third aspect of the invention, a digital system on a semiconductor chip includes a central processing unit (CPU) coupled to a first bus in a first clock domain defined by a first bus clock frequency; a plurality of electronic devices coupled to a

30   second bus in a second clock domain defined by a second bus clock frequency; a bus bridge coupled between the first and second buses and operable to allow communication

between the CPU at the first bus clock frequency and one of the plurality of electronic devices at the second bus clock frequency; a programmable logic device (PLD) coupled to a third bus in a third clock domain; and a PLD bridge coupled between the second and third buses.

5   The following detailed description and the accompanying drawings provide a better understanding of the nature and advantages of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is diagram of a digital system with a programmable logic integrated

10   circuit;

FIG. 2 is a block diagram of a digital system according to an embodiment of the present invention;

15   FIG. 3 is a block diagram of a system having a multiple bus architecture according to an embodiment of the present invention;

FIG. 4 shows a more detailed and exemplary diagram of a first bus in FIG. 3, and its connectivity to exemplary components and peripherals, according to an

20   embodiment of the present invention;

FIG. 5 shows a more detailed and exemplary diagram of a second bus in FIG. 3, and its connectivity to exemplary components and peripherals, according to an embodiment of the present invention; and

25

FIG. 6 shows an exemplary block diagram of a bridge according to an embodiment of the present invention.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

30   FIG. 1 shows a block diagram of a digital system within which the present invention may be embodied. The system may be provided on a single board, on multiple boards, or even within multiple enclosures. FIG. 1 illustrates a system 10 in which a programmable logic device 106 may be utilized. Programmable logic devices are

4

currently represented by, for example, Altera's MAX®, FLEX®, and APEX™ series of PLDs.

In the particular embodiment of FIG. 1, a semiconductor device 100 is coupled to a memory 102 and an I/O 104 and comprises a programmable logic device (PLD) 106 and embedded logic, which may include, among other components, a processor 109. The system may be a digital computer system, digital signal processing system, specialized digital switching network, or other processing system. Moreover, such systems may be designed for a wide variety of applications such as, merely by way of example, telecommunications systems, automotive systems, control systems, consumer electronics, personal computers, and others.

Referring now to FIG. 2, there is shown a diagram of a system 20 having a multiple bus architecture, according to an embodiment of the present invention. The bus architecture is comprised of bus masters 200, 201, 202 and 204, each of which can communicate with one or more of the peripherals in the system, e.g., memory 206, and other peripherals 208-216 such as, for example, I/O devices, etc., via bridges 218-224. The principle function of each bus master is to manage the bus it is associated with and control what devices can access the bus. Bridges 218-224 function to allow communication between a bus master in a first clock domain with a peripheral in a second clock domain, thereby allowing components on each bridge to operate at their individually optimal speeds. A bridge accomplishes this by preferably including a first-in first-out (FIFO) buffer, which accepts data at the clock rate of a first bridge and writes it out to a second bus at the clock rate of the second bus. So long as each bus master is accessing a different peripheral on a different bus, employment of bus bridges 218-224 leads to enhanced system performance, since multiple bus masters can communicate with different peripherals on different buses simultaneously without the problem of bus access contention. In other words, this embodiment of the present invention provides for the division of processing elements into their own clock domains 226-232 and provides bridges 218-224, which allow communication to other devices on buses across clock domains 226-232. Nevertheless, the bus architecture of system 20 is flexible enough to accommodate multiple bus masters, e.g. bus masters 200 and 202, sharing the same bus. The only condition is that the bus masters run at the same frequency. Each clock domain can derive from independent clock sources or derive from a division of one or more clock

sources. Whereas the embodiment in FIG. 2 is shown to have a certain number of bus masters and peripheral devices, it should be realized that this number is merely exemplary and that a design having any number of bus master, buses, bridges and peripherals is possible and, therefore, within the scope of the present invention.

5

FIG. 3 shows a portion of embedded logic illustrating an exemplary implementation of the multiple bus architecture shown in FIG. 2. Access to a peripheral is controlled by a number of bus masters connected by a bus structure comprised of two or more buses, and which is described in greater detail below. In this exemplary

10   implementation, there are three bus masters, including processor 300, PLD Master 302 and Configuration Logic 304. These bus masters 300-304 are capable of initiating read and write operations by providing address and control information. Processor 300 is connected to a first bus 306 (e.g. a 32-bit AHB bus). First bus 306 also connects to one or more peripheral devices such as a synchronous dynamic random access memory

15   (SDRAM) controller 330, on-chip static random access memory (SRAM) (single 310 and dual 312 port), processor only peripherals, for example, an interrupt controller 314 for receiving an interrupt signal from another peripheral and reporting the signal to the processor 300, and a watchdog timer 316, which functions to cause the system to reset if, for example, certain logic states within processor 300 do not toggle within a predefined

20   time period. A test interface controller (TIC) 318 can also be connected to first bus 306 for functional testing.

The remaining bus masters, which in this example are PLD Master 302 and Configuration Logic 304, share a second bus 307. Second bus 307 can be, for

25   example, a standard 32-bit AHB bus that can provide for a lower memory access speed, by PLD Master 302 and Configuration logic 304, than may be required for processor 300, which is, as described above, connected to first bus 306. Similarly, peripherals that can be accessed with a relatively larger degree of latency tolerance can be connected to second bus 307. Some of the modules connected to second bus 307 may include, for

30   example, a universal asynchronous transceiver (UART) 320, a bus expansion 322, a timer 324, clock generator 326, a reset/mode controller 328, an SDRAM memory controller 330 for controlling external SDRAM, and single and dual on-chip static random access memories (SRAMs) 310 and 312. Bus expansion 322 is used primarily to connect to external memory, for example, Flash memory from which processor 300 can boot. Clock

generator 326 is preferably programmable so that a desired clock frequency can be set for second bus 307. Both single 310 and dual 312 SRAMs may be divided into multiple blocks (e.g. divided in two, as in FIG. 4), each having their own bus arbitration. Division permits concurrent access to different blocks by bus masters on first 306 and second 307

5   buses. Second bus 307 is also connected to a PLD slave bridge 332 and a PLD master bridge 334, each of which is interfaced to a PLD in the system (not shown in FIG. 3), via third 336 and fourth 338 buses, respectively. Third 336 and fourth 338 buses can be, for example, standard 32-bit AHB buses. (Alternatively, a bridge to and from the PLD may be configured in a single device.) In this particular embodiment, the PLD may be, for

10   example, an APEX™ 20KE, which is manufactured by Altera Corporation and described in *Altera Data Book* (1999), which is incorporated by reference.

FIG. 4 shows first bus 306 in greater detail. First bus 306 is clocked by, for example, a dedicated phase locked loop (PLL), which allows the maximum possible

15   performance to be achieved by processor 300. The clock frequency can be made selectable by writing to clock generator module 326. An address decoder 440 provides selection of bus bridge 325, SDRAM memory controller 330, on-chip SRAM 310 and 312, interrupt controller 314 and watchdog timer 316 in accordance with memory maps of the various modules. Address decoder 440 selects one of these elements by comparing

20   address information encoded in memory map registers (not shown in FIG. 3) on second bus 307 to an address output by processor 300. If the address output by processor 300 is within an address range of any one of the elements on first bus 306, then a select line for the corresponding element is activated. If access is not being made for elements coupled exclusively to first bus 306 (e.g. memory controller 330, interrupt controller 314,

25   watchdog time 316) or for SRAM 310 or 312, then access is directed to an element on second bus 307 via bus bridge 325.

FIG. 5 shows second bus 307 from FIG. 2 in greater detail. Second bus 307 may be clocked by, for example, a divided down version of the clock that clocks first

30   bus 306 or may be a clock unrelated to the first bus clock. A register for selection of this frequency is located within clock generator module 326. Address decoder 340 provides for selection of SDRAM memory controller 330, bus expansion 322, on-chip SRAM 310 and 312, UART 320, clock generator 326, timer 324, reset/mode control 328, PLD slave

bridge 332, etc. according to the system's memory map. Reset/mode controller 328 functions to reset the system and control its mode of operation. It may also contain memory map registers a user can access to configure a memory map for the system. Second bus 307 also includes an arbiter 542 for determining which bus master, PLD master 302 or configuration logic 304 or bus masters on first bus 306 (via bus bridge 325) has access to second bus 307.

First 306 and second 307 buses are coupled to each other by bus bridge 325. PLD master 334 and slave 332 bridges are substantially identical with bus bridge 325 with only minor differences related to the chosen address decoding scheme and bus structure. An exemplary embodiment of a bridge 60 is shown in FIG. 6. An originating bus 600 of a transaction is connected to that bridge's slave 602 while that bridge's master 604 is connected to a destination bus 606. Bridge 60 includes synchronization logic 608, which allows the master and slave interfaces to reside in different clock domains. The master and slave interfaces of bridge 60 can be synchronous or asynchronous relative to each other. If synchronous, bridge 60 can be configured to bypass synchronization logic 608 to reduce the latency through bridge 60.

A write buffer 610 is configured to accept bursts of posted write data from slave interface. Preferably, the bus protocol allows for several transfers of write data to be concatenated to enhances bus performance. No wait states are inserted so long as a buffer entry is free to accept the data. A write request is generated by slave interface and is synchronized to the master clock domain. Master 604 de-queues data from write buffer 610, writes it out to destination bus 606 and then asserts an acknowledge signal to slave 602 to indicate that a buffer entry is now free for re-use by slave 602. Sending an acknowledge signal back to slave 602 accounts for the difference in clock frequencies in the slave and master clock domains. Without write posting, for example, if master 604 is processor 300 on first bus 306 and slave 602 is one of the slaves on second bus 307, as in FIG. 3, processor would have to wait for each single transfer to complete before it send the next transfer. Since processor 300 will normally run at a higher frequency than slaves on second bus 307, write posting allows the processor 300 to run at its optimal speed. In an exemplary embodiment, write posting is controlled by action of the bridge coupled between the two buses. Preferably, each bridge includes a first-in first-out (FIFO), which accepts data at the clock rate of the first bridge, buffers it and writes it out to the second

8

bus at the clock rate of the second bus. The FIFO thereby allows processor 300, for example, to carry out its next action at its own optimal clock rate and is not stalled by having to wait for data to be written to the second bus 307.

5   When selected by a read transaction, slave 602 asserts a read request that is synchronous to the master clock domain. Master 604 performs a read transaction (pre-fetching data to fill a read buffer 612 if enabled) and asserts an acknowledge signal to indicate when data is available. Read buffer tags are used to return the status of the transaction (e.g. OK, ERROR, RETRY).

10

Slave interface also provides access to a bridge status register and address status register (not shown in FIG. 6). These registers contain information pertaining to a posted write transaction that resulted in an ERROR response, could not arbitrate for the destination bus, or could not complete an access that had a RETRY response. When

15   slave 602 indicates that a transfer is pending, master 604 uses the address and control information to perform the requested transaction on destination bus 606. Master 604 will only read data from destination bus 606 if there is a free entry in read buffer 612 to receive it. If no free entries are available, then master 604 will insert BUSY cycles. Similarly, if no data is available from write buffer 610 during a write transaction, master

20   604 will insert BUSY cycles.

In conclusion, the present in invention discloses a bus architecture of the present invention, in particular, is embodied as a multiple bus master system, which allows communication among all peripherals in the system via bridges that de-couple

25   clock frequencies of the individual bus masters from the peripheral they are accessing. The bus architecture of the present invention, therefore, allows various system units to run at their optimal speeds and reduces bus contention.

The foregoing description of preferred exemplary embodiments has been

30   presented for the purposes of description. It is not intended to be exhaustive or to limit the invention to the precise form described herein, and modifications and variations are possible in light of the teaching above. Accordingly, the true scope and spirit of the invention is instead indicated by the following claims and their equivalents.